

PIGLE - Particles Interacting in Generalized Langevin Equation simulator

N Avidor^{a,1,*}, PSM Townsend^{a,b,1}, DJ Ward^{a,1}, AP Jardine^a, J Ellis^a, W Allison^a

^a*Cavendish Laboratory, University of Cambridge, Cambridge CB30HE, United Kingdom*

^b*Current Address: Rutgers University, Department of Chemistry and Chemical Biology, 123 Bevier Road, Piscataway, NJ 08854, United States*

Abstract

We present a package using Simulink and MATLAB to perform molecular dynamics simulations of interacting particles obeying a Generalized Langevin Equation. The package, which accounts for three spatial dimensions and rigid-body like rotation, is tuned to explore surface diffusion of co-adsorbed species. The physical parameters are species specific, and include user-defined colored noise spectra and memory friction kernels acting independently on translational and rotational degrees of freedom. We benchmark the simulations using established analytical results for dynamical correlation functions, and we use the package to numerically verify novel analytical results concerning dissipative rotational motion and multi-exponential friction kernels. The package provides a straight-forward way to expand the modeling of ultra-fast surface diffusion problems at the atomic scale.

Keywords: PIGLE, Molecular Dynamics, Spin Echo, Surface Science, Diffusion, Langevin Equation

1. PROGRAM SUMMARY

Program title: PIGLE - Particles Interacting in Generalized Langevin Equation simulator

Licensing provisions: GNU General Public License 3 (GNU/GPL 3.0)

*Corresponding author

Email address: `na364@cam.ac.uk` (N Avidor)

¹Equal contribution

Programming language: MATLAB/Simulink

Supplementary material (if any):

Nature of problem: Understanding molecular diffusion at surfaces is of fundamental and industrial interest, and requires bridging between the experiment and ab-initio theories. Molecular dynamics simulations are widely used to bridge the experimental and theoretical worlds, however normally surface dynamics simulations are done either under extreme simplifications (two dimensional Langevin with for single species adsorbate), or with all the substrate atoms as well as the adsorbate atoms included explicitly - at expensive computational cost.

Solution method (approx. 50-250 words): We here present a simulator for generalized Langevin dynamics of co-adsorbed interacting species, accounting for surface diffusion in 3 spatial dimensions and rigid body rotation. The package allows also to perform statistical analysis of the computed trajectories, considering a configuration of scattering centers which is imposed on the translating and rotating center-of-mass.

Additional comments including Restrictions and Unusual features (approx. 50-250 words): References:

2. Introduction

Understanding molecular diffusion at surfaces is of fundamental importance in water studies, catalysis, atmospheric chemistry and in other fields.[1, 2, 3, 4, 5] Exploring surface-dynamics at near-desorption and desorption temperatures, which are of technological importance, normally requires the use of spectroscopic techniques, as the motion is too fast to image in real-space. One spectroscopic technique to study surface dynamics at elevated temperatures is Helium-3 Spin-Echo (HeSE),[6, 7] which uniquely allows the study of surface dynamics at sub-nanometer length-scale and picosecond to nanosecond time-scale. During its first decade of operation, the technique has been used mainly to observe surface diffusion of atoms and some small molecules. Due to the loss of phase information in spectroscopic studies,[8] it is normally impossible to directly infer the exact trajectories of particles at the surface. Hence, analytic models and simulations are normally used to obtain physically meaningful insights from the experimental results.[9]

One of the most common methods for modeling surface dynamics is to use Langevin equation based molecular dynamics (MD) simulations,[9, 10] in

which the interaction of adsorbates with the substrate atoms is described using a frozen potential energy surface (PES), a drag term and a stochastic fluctuating force. As a result, meaningful insights into the adsorbate-substrate interaction can be obtained. In comparison with full (micro canonical) MD simulations, which include the interactions of all atoms, Langevin based MD is more economical in terms of computing time, while offering a comparable quantitative description of experiment.[10] The Langevin equation with a simple drag coefficient and white noise can arise due to non-adiabatic coupling to very rapidly fluctuating electronic degrees of freedom. However, first-principles predictions of the electronic friction coefficient show that electronic dissipation is not always sufficient to reproduce the friction required to explain HeSE data.[11] The remaining potential sources of dissipation, such as phonons, are in general likely to cause memory effects. These can be described with and motivate the use of a Generalized Langevin solver, as implemented in the present work.

Use of Langevin dynamics simulations has become a standard practice in analysis of surface-dynamics HeSE studies. Traditionally however, the diffusing species were considered to be point particles responding to a two-dimensional (2D) PES. Furthermore, simulations were performed and analyzed only for ensembles consisting of a single species of adsorbates. In recent years, pioneering works to study adsorption of larger molecules have demonstrated the need to consider dynamical degrees of freedom beyond translational diffusion in a 2D PES.[12, 13]

Here we describe PIGLE (Langevin Based Molecular Dynamics simulator), a package to simulate co-motion of interacting particles from multiple species in 2D to 4D PES (i.e. up to three spatial degrees of freedom, and rigid body rotations). The simulation allows consideration for colored-noise driven friction (as explained below), separately for translation and rotation. The structured core simulation can be expanded to address problems beyond 4D, to allow exploration of how spectral properties of friction and noise affect motion in higher dimensional PES. In that respect, PIGLE complements long-time simulations including methods such as *Nudged Elastic Band* (NEB) and the *String Method*, which are normally used to find minimum-energy transition path in a given PES.[14, 15]

One particular use of the core MD simulator in PIGLE, can be to support scattering calculations from moving adsorbates at surfaces. Specifically, To support analysis of measurements obtained with techniques such as HeSE, the package allows calculation of the intermediate scattering function, either

coherent or incoherent. The package includes routines to expand centre-of-mass and angular trajectories into trajectories for the complete collection of point objects making up every rigid body in the simulation, which can be used either for visualization purposes or to calculate the scattering functions assuming every point particle acts as a scattering centre with equal contributions. The role of rotations in the surface diffusion process can therefore be conveniently studied. Further, the effect of rotations on measured scattering functions in helium atom scattering experiments can be modeled in a way that reflects the degree of inhomogeneity of the electron density in the adsorbed species.

In the present work, we express the Generalized Langevin Equations in a form suitable for solution by standard methods encountered, for example, in digital signal processing. Our approach, therefore, enables us to exploit the optimized code and graphical programming environment offered by Simulink and MATLAB.

3. Underlying Equation

PIGLE solves the Generalized Langevin Equation (GLE) to calculate the dynamical evolution of a mutually interacting adsorption system. However, the Langevin Equation (LE) has been much more widely used in surface diffusion studies up to now,[7, 6] and therefore it is useful to review the approach.

The Langevin Equation. Consider an adsorption system in which there are P homogeneous adsorbate populations (labelled $\mu=1:P$). Each adsorbate in a given homogeneous population interacts with the substrate via the same single-particle potential $V_\mu(\mathbf{x})$ and microscopic drag coefficient, and experience inter-adsorbate forces with particles from all populations. The 3D translational motion of the i^{th} particle in homogeneous μ^{th} population of adsorbates can then be described using the Langevin equation

$$m\ddot{\mathbf{x}}_i(t) = -\nabla V_\mu(\mathbf{x}_i(t)) - m\gamma_\mu\dot{\mathbf{x}}_i(t) + \boldsymbol{\xi}(t) + \sum_{j \neq i} \mathbf{F}_{ij}, \quad (1)$$

with γ scaling the isotropic drag, a stochastic force $\boldsymbol{\xi}$ which represents the fluctuations, and an inter-adsorbate interaction force (j runs over all the particles in the simulation). At present, interactions in PIGLE are restricted to radial pairwise forces, which although not completely general, can represent

a rich variety of structural and dynamical behaviour. To satisfy the classical fluctuation dissipation theorem,[16] the kinetic energy introduced by ξ is balanced by the frictional term, such that on average, thermal equilibrium is maintained at constant temperature T . In order to do so, the *dimension specific* stochastic force is constructed to have the autocorrelation function

$$\langle \xi(t)\xi(t') \rangle = 2m\gamma k_B T \delta(t - t'), \quad (2)$$

where the angle brackets refer to the inverse Fourier transform of the power spectrum, which defines an autocorrelation function according to the Wiener-Khinchin theorem. The random force ξ is taken typically as Gaussian distributed with zero-mean.[17] Furthermore, there are no correlations between the fluctuating force on one co-ordinate (dimension) and the fluctuating force on another.

The Generalized Langevin Equation. In PIGLE, the dynamics are described using the GLE,[18, 19] which allows the random impulse term to have a non white noise characteristic. Filtering the noise provides a mechanism to represent a physically reasonable spectral density of the coupling between the adsorbate and excitations in the substrate heat bath. For example, a low-pass filter will enhance low frequency impulses, which may be a better representation of the noise from thermal phonons at the physical surface. The GLE can be written for the i^{th} particle as

$$m_i \ddot{\mathbf{x}}_i(t) = -\nabla V_\mu(\mathbf{x}_i(t)) - m_i \int_{-\infty}^t \gamma_i(t-t') \dot{\mathbf{x}}_i(t') dt' + \zeta_i(t) + \sum_{j \neq i} \mathbf{F}_{ij}. \quad (3)$$

For spatial dimensions, m_i is the mass of the particle, and γ_i is the isotropic translational friction. For angular dimensions, \mathbf{x} is replaced with θ (rotational co-ordinate), m_i with the angular mass of the i^{th} particle, and γ with $\gamma_{Angular}$, the friction for the rotational motion. The friction kernel $\gamma_i(t)$ is related to the noise ζ by

$$\langle \zeta_i(t)\zeta_i(t') \rangle = 2m_i k_B T \gamma_i(t - t'), \quad (4)$$

which guarantees a constant temperature T by the fluctuation-dissipation theorem.

Direct numerical solution of the GLE as formulated in Equation 3 is computationally expensive, as at every time-step, the entire history of the particle must be included in the integral over the friction kernel. Here we use

a method suggested by Ceriotti [20] that allows us to express the GLE using a standard engineering approach for a discrete, state-space method.[21, 22] For each particle in the simulation (with momentum p), and for each dimension x of the simulation, the GLE is represented as a set of differential equations, which depend only on the current time t :

$$\begin{aligned} \dot{x}(t) &= \frac{p(t)}{m}; \\ \begin{pmatrix} \dot{p}(t) \\ \dot{\mathbf{s}}(t) \end{pmatrix} &= \begin{pmatrix} -V'(x(t)) - \sum_{j \neq i} F_{ij}(t) \\ \mathbf{0} \end{pmatrix} - \mathbf{A} \begin{pmatrix} p(t) \\ \mathbf{s}(t) \end{pmatrix} - \mathbf{B}\boldsymbol{\xi}(t). \end{aligned} \quad (5)$$

The matrices \mathbf{A} and \mathbf{B} with dimensions $(n+1) \times (n+1)$ govern the dissipation and fluctuations, and can represent the filter applied in the friction kernel. A set of filters is provided with PIGLE, and a more detailed explanation can be found in Ref [20]. $\mathbf{s}(t)$ then is an n -elements vector with units of momentum and $\boldsymbol{\xi}$ is a vector of $(n+1)$ white noise sources. The fluctuation-dissipation theorem, which relates the noise spectrum and the friction kernel, is enforced by requiring that \mathbf{A} and \mathbf{B} are related by

$$mk_B T (\mathbf{A} + \mathbf{A}^T) = \mathbf{B}\mathbf{B}^T. \quad (6)$$

4. Main Components of the Code

PIGLE is intended to model the dynamics of co-adsorbed populations of adsorbates on periodic surfaces in situations where the coverage (or more precisely, number density defined as number of adsorbates per substrate atoms) is finite and possibly variable. To represent a finite coverage, the code employs a super-cell approach, which is a common feature of several molecular dynamics codes.

Implementation in Simulink. Simulink[23] is a package by MathworksTM which allows graphical coding for simulations of time-dependent systems. The simulation code is represented in a block diagram, with information flows (in the form of lines) from output ports of one block to the in-port(s) of the other block(s). Blocks are the equivalent of functions in traditional programming languages, and each block can contain subsystems and model references (simulations which are included in another simulation) as part of

its block diagram. Simulink was chosen due to the convenience of its close interaction with MATLAB code, while also providing compilation options to increase the simulation speed. Furthermore, Simulink offers a relatively straight-forward path to deploy the code on field-programmable gate arrays (FPGA), which can potentially achieve a speed-up of code execution of more than an order of magnitude. Simulink is widely used in digital signal processing and automatic control.

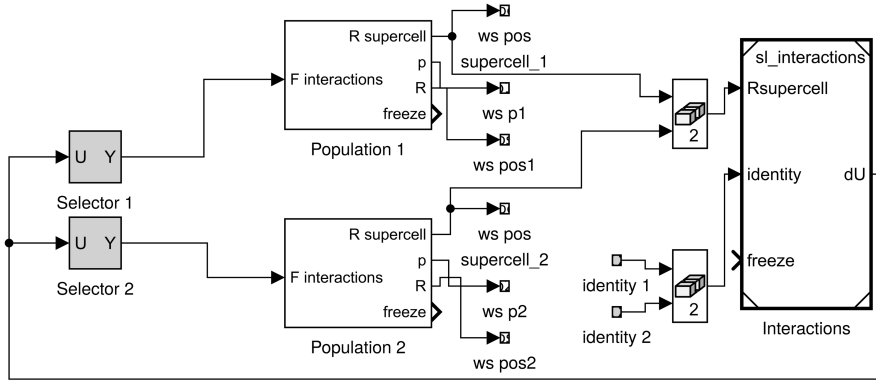


Figure 1: The top-level Simulink block of the PIGLE simulation. In each time-step, the *Interactions* block calculates and outputs the inter-adsorbate forces acting on all particles. *Selector* blocks select the force-information for all particles pre-assigned to a specific particle-population, which is then fed to the population dedicated *Population* blocks. Each *Population* block integrates the GLE for a single time-step and outputs the new momentum and position (total and super-cell relative), to the MATLAB workspace, and for calculation of inter-adsorbate forces. The position from all particle populations is combined and fed to the *Interactions* block, where the inter-adsorbate forces are computed for the next time-step. The inputs *identity 1,2* to the *Interactions* block hold the identity (species) of each particle.

In the following exposition, we aim to set out the basic computing blocks, in rough outline, as an illustration of the way in which PIGLE is constructed. The code is derived from a hierarchical structure of computational blocks, which we describe below, starting at the highest level.

Figure 1 shows the highest level block, tailored to simulate co-adsorption of two different species (*Populations*). In each time-step, given information on forces from inter-adsorbate interactions, each of subsystems *Population 1* and *Population 2* calculate the new positions and momenta of particles which were assigned to it. Such a group of particles shares all the input parameters

for a GLE simulation of a homogeneous population, except of the inter-adsorbate forces. Each of the *Population* subsystems outputs the position (relative to the origin, and to the origin of the super-cell) and the momentum of the constituent particles to the MATLAB workspace for recording. The position in the super-cell is also passed to the *Interactions* block, where the inter-adsorbate forces are computed. The forces are then fed back for the computation of motion in the next time-step .

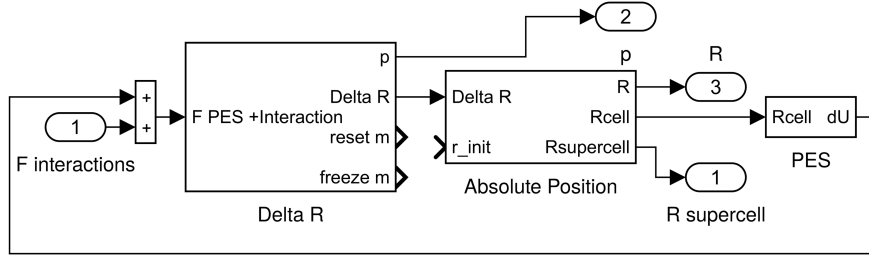


Figure 2: Block to compute the motion of population-specific particles. At each time-step, the inter adsorbate force on each particle is fed from the in-port, and is combined with the force resulting from the PES. The *Delta R* block then computes the momentum which is passed to an out-port, and the total change in translation/rotation which is passed to the *Absolute Position* block. It is there where the initial position is added to obtain the absolute position of each particle. The position within the simulation super-cell is calculated, and the two quantities are passed to out-ports. Position within the unit cell is also calculated, and is passed to the PES block for calculation of the PES-resulted force.

An example of a specific *Population* block, a block which calculates the motion for particles of one population, is presented in figure 2. Inter-adsorbate interaction forces, which are supplied as an input to the *Population* block, and forces from the *PES* subsystem, are combined at each time-step and are then fed into the *Delta R* subsystem. The *Delta R* subsystem computes the momenta of all particles, and their displacements from their positions at $t = 0$ by integration. The integration method can currently be set to *Explicit Euler* or *Trapezoidal* based. While the momentum is passed to the out-port of the *Population* block, the net position is passed to the *Absolute Position* subsystem. The *Absolute Position* subsystem adds the initial position to the net trajectory, to obtain the absolute position of each particle. It then calculates the position within the unit cell and the super-cell of the simulation, and outputs the absolute position and super-cell relative position to the upper block-level. The unit cell relative position is passed to the *PES*

subsystem, where the force from the *PES* is calculated. Information from the *PES* subsystem is fed back for usage in the next time-step.

The *Delta R* block contains *gle* subsystems (one for each enabled dimension) which integrates the GLE equation for each particle. Figure 3 presents an example subsystem, which is based on the work of R. Collingham.[24]. In each time-step, the forces resulting from two subsystems, the *PES* and inter-adsorbate interactions, are fed to the *gle* in-port dU . The sum of dU , the drag force and random impulse is then integrated and is both fed back to the frictional term, and is propagated for further calculations (i.e force from *PES* and other adsorbates at the next time-step).

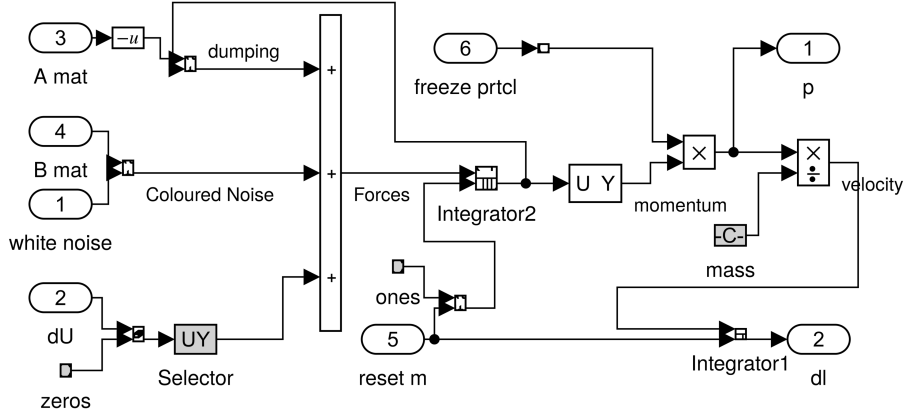


Figure 3: A subsystem to solve the GLE for a group of particles in one dimension. Four inputs are the population specific \mathbf{A} and \mathbf{B} matrices (Equation 5), white noise source, and the sum of the forces derived from inter-adsorbate interactions and the gradient of the single-adsorbate PES. The summation of the three components in the right hand of Equation 5 gives the time derivative of the momenta. A discrete integrator block is used to obtain the momenta (a combined vector of p and s from Equation 5). The momenta are then both fed back to be multiplied with the matrix \mathbf{A} , and passed through a selector which selects the momentum p (the first element in the momenta). The position of particles is obtained via further integration (which can currently be based on *Explicit Euler* or *Trapezoidal* methods), and both the momentum and 1D position are passed to out-ports. The same \mathbf{A} and \mathbf{B} matrices are used for spatial dimensions, but different ones are being used for rotational dimensions. Equally, for the angular dimension, an angular mass is being used.

To calculate interactions between particles, the simulation assumes a super-cell. The nature of the super-cell is determined in the MATLAB wrapper, and is a combination of the unit cell, the user-defined adsorbate coverage,

and the user-defined total number of adsorbed particles. The position of the particles in the super-cell is used then to compute the inter-adsorbate interactions. PIGLE currently assumes a pair-wise, isotropic interaction between the centers of mass, and no rotational interaction. The interaction between a particle i and a neighbor j is calculated using the *Minimum Image Convention*; shadow j particles are assumed in adjacent (eight) super-cells around the actual super-cell. Out of the total nine j particles which surrounds particle i , the interaction is computed between the pair i - j with the shortest distance.[25, 26] An implicit assumption is that the maximum distance for interactions is half of the super-cell size, so adsorbate i can interact with its neighbor j (in the same super-cell), or shadow neighbor j from adjacent super-cell, but not both. In future work a connection list of particles will be implemented, which is expected to reduce the scaling of computational time used in the interaction block from quadratic to linear with the total number of particles, N .

Both inter-adsorbate and adsorbate-substrate interactions, as well as other parameters are pre-defined in a structure (constructed by the MATLAB wrapper) which is used by the model. The simulation logs trajectories and momentum for particles in each population.

5. MATLAB Wrapper and additional functionality of the PIGLE package

The MATLAB wrapper which is provided with PIGLE, `run_pigle.m`, performs three principal tasks: (a) it configures the Simulink model, (b) it executes the Simulink simulation, and, if required by the user, (c) it calculates the coherent and incoherent intermediate scattering function (ISF) which can be compared with scattering experiments.[27]

Configuring the model. After initial tasks, the file `config_model.m` is called. In this file one can define general parameters of the simulation, such as number of steps, total simulation time, thermalization time, momentum transfer ($\Delta\mathbf{K}$) values and the ISF is to be calculated (Coherent, Incoherent, both, if at all). A `surface_params.m` file provides all the surface related parameters (lattice properties, adsorbate properties, parameters for inter-adsorbate interactions). The values of $\Delta\mathbf{K}$ where the ISF is calculated, are rounded onto a grid whose points are determined by the super-cell dimensions (the grid is specified precisely later in the present section). Further, the particles

are given uniform random initial positions, and momenta either thermally distributed (default) or zero (optional). Built-in into the scripts are auxiliary functions, allowing for example the generation of \mathbf{A} and \mathbf{B} matrices for the GLE, enabling specification of different memory kernels (applying different filters). Similarly, the PES can be generated given several principal parameters.

Executing the model. In generating the intermediate scattering function, a statistical average is achieved by executing the simulation repeatedly (for a requested number of iterations) and averaging over the resulting ensemble. Using the *parsim* command, the iterations can be computed in parallel, taking into account the maximum number of available workers for parallel computing. The model can be executed using rapid-acceleration mode, which provides compilation of the model before execution.

Calculation of the Intermediate Scattering Function (ISF). Following completion of a simulation, the trajectories can be used to calculate the (coherent and/or incoherent) ISF. The rotational co-ordinate, and the spatial co-ordinates of the center of mass, are used to define a particular molecular configuration for the diffusing particles. For example, the user can define a ring with six particles, and the MATLAB-wrapper will expand the trajectory of each adsorbate (of the relevant population) to six trajectories according to the position of the center of mass, and the rotational co-ordinate. Given a vector of amplitudes for each particle i , $A_i = \exp(-i\Delta\mathbf{K} \cdot \mathbf{r}_i)$, where $A_i(t_\nu)$ is the element for time-step ν , the incoherent scattering function from an adsorbate i , $S_i(\Delta\mathbf{K}, \Delta\omega)$, can be calculated as

$$S_i(\Delta\mathbf{K}, \Delta\omega) = \mathcal{F}\{A_i\}^* \cdot \mathcal{F}\{A_i\}, \quad (7)$$

where \mathcal{F} marks a Fourier transformation and $\hbar\Delta\omega$ the energy exchange. The overall incoherent scattering function is the sum $S_{inc} = \sum_i S_i$ over the particles in all populations. The coherent scattering function is calculated by first summing the amplitudes of scattering from all particles, $A = \sum_i A_i$, and then computing the scattering function, $S_{coh}(\Delta\mathbf{K}, \Delta\omega) = \mathcal{F}\{A\}^* \cdot \mathcal{F}\{A\}$. The coherent and incoherent ISF are calculated by inverse Fourier transformation of S for each iteration, and the final ISFs are the mean over all the iterations.

Adjustment of momentum transfer points for calculation of ISFs. It is assumed that a particle and its shadow particles (from adjacent super-cells) scatter the helium probe in-phase. This assumption is essential, since while the simulation assumes shadow particles, the calculation of the ISF does not take those particles explicitly into account. Given a scattering amplitude from an adsorbate $A_\mu = \exp[-i(\mathbf{R} \cdot \Delta\mathbf{K})]$, the condition for in-phase scattering of the adsorbate and its shadow particles can be written as

$$\exp[-i(\mathbf{R} + \mathbf{a}) \cdot \Delta\mathbf{K}] = \exp[-i\mathbf{R} \cdot \Delta\mathbf{K}] \Rightarrow 2\pi n = \Delta\mathbf{K} \cdot \mathbf{a}, \quad (8)$$

where \mathbf{R} is the particle position in the super-cell, and \mathbf{a} is a super-cell lattice vector. The user-defined values of the momentum transfer are rounded on to a grid of points satisfying the in-phase condition 8, i.e. the reciprocal lattice vectors associated with the simulation super cell.

Visualization. The MATLAB wrapper `run_pigle.m` provides plotting functions for the calculated ISFs. For further visualization, a script `make_movie.m` is provided.

6. Benchmark calculations

We provide several examples to demonstrate and illustrate the performance and capabilities of the code. We assess the validity of the results via comparison of the computed ISF(s) to analytic models where available, and to qualitative expectations and previous numerical results in cases where analytic models are not available. As well as the experimental connection with scattering experiments, there are conceptual advantages to quantifying the adsorbate dynamics using the ISF. It is well known that the ISF, as the temporal fourier transform of the scattering function as defined in Equation 7, is also the spatial Fourier transform of van Hove’s conditional probability function.[28] The ISF therefore contains essentially complete statistical information on the adsorbate dynamics.

6.1. Analytic ISFs

6.1.1. Translational GLE

A key feature of the PIGLE package is the facility to flexibly and efficiently explore different forms of memory friction, in which the friction kernel is naturally represented as the sum of exponential functions in the time domain. To validate the implementation of memory friction in translational

motion, we perform simulations of the single-particle GLE in a flat potential [$V(\mathbf{R}) = 0$]. In Figure 4, simulation results (circles, squares and diamonds markers) are confirmed against analytical ISFs from the GLE with three different friction kernels (solid, dashed and dotted curves, labelled in the figure). In one test (dotted line), the friction kernel is a mono-exponential function $\gamma(t) = \gamma\omega_c \exp(-\omega_c t)$ where the cutoff frequency ω_c is sufficiently high that memory effects represent a minor quantitative correction to Langevin dynamics. In the second (dashed line), another mono-exponential kernel is tested in which ω_c is low enough to induce oscillatory behavior in the ISF. In the third comparison (solid line), the friction kernel is taken as the sum of the previous two kernels, forming the bi-exponential kernel

$$\gamma(t) = \gamma_1\omega_1 \exp(-\omega_1 t) + \gamma_2\omega_2 \exp(-\omega_2 t). \quad (9)$$

Further details of the simulation parameters are provided in the caption of Figure 4.

The analytical ISF in the case of a mono-exponential kernel has been derived previously using an exact second-order cumulant expansion relating the ISF to the velocity autocorrelation function (VACF) $\psi(t)$:

$$I(\Delta K, t) = \exp \left[-\Delta K^2 \int_0^t dt' (t-t') \psi(t') \right]. \quad (10)$$

The VACF and the friction kernel are compactly related in Laplace space. Defining $\Psi(s)$ and $\Gamma(s)$ as the Laplace transforms of $\psi(t)$ and $\gamma(t)$ respectively, the transforms are related by [29, 30]

$$\Psi(s) = \frac{k_B T / m}{s - \Gamma(s)}, \quad (11)$$

which can be analytically inverted to obtain $\psi(t)$ for sufficiently simple forms of $\Gamma(s)$. When $\gamma(t)$ is a mono-exponential function $\gamma(t) = \gamma\omega_c \exp(-\omega_c t)$, the inversion to obtain $\psi(t)$, and the subsequent integration of $\psi(t)$ to obtain the exponent of Equation 10 produces [30]

$$I(\Delta K, t) = \prod_{k=1}^2 \exp \left(-\Delta K^2 \frac{k_B T}{m} \frac{p_k}{s_k^2} \left[e^{s_k t} - s_k t - 1 \right] \right), \quad (12)$$

where the two s_k are the solutions of a quadratic equation $s^2 + \omega_c s + \gamma\omega_c = 0$, and the corresponding p_k are given by

$$p_1 = \frac{(s_1 + \omega_c)}{s_1 - s_2} ; p_2 = \frac{(s_2 + \omega_c)}{s_2 - s_1}. \quad (13)$$

Equation 12 constitutes the solid curves in Figure (4) for the two mono-exponential kernel tests. The method of Laplace transforms followed by a cumulant expansion is readily extended to cover multi-exponential kernels, except that the roots s_k are the solutions of a polynomial of higher degree. For the bi-exponential kernel (9), the ISF is given by

$$I(\Delta K, t) = \prod_{k=1}^3 \exp \left(- \Delta K^2 \frac{k_B T}{m} \frac{p_k}{s_k^2} \left[e^{s_k t} - s_k t - 1 \right] \right), \quad (14)$$

where the three s_k are the solutions of the cubic equation

$$s^3 + (\omega_1 + \omega_2)s^2 + (\omega_1\omega_2 + \gamma_1\omega_1 + \gamma_2\omega_2)s + (\gamma_1 + \gamma_2)\omega_1\omega_2 = 0, \quad (15)$$

and the corresponding p_k follow the same pattern as seen in the mono-exponential kernel result:

$$p_k = \frac{(s_k + \omega_1)(s_k + \omega_2)}{(s_k - s_{k'})(s_k - s_{k''})}, \quad (16)$$

where $k' \neq k'' \neq k$. Equation 14 is included as a function in LAMPD, which has been used to plot the solid curve in Figure 4, in agreement with the simulated data. The cubic equation 15 is solved numerically. The solution of progressively higher degree polynomial equations is the only aspect of the method requiring numerical treatment.

6.2. Rotational ISF

In PIGLE, the rotational co-ordinate of each rigid body is treated on the same footing as translational co-ordinates, in that it obeys GLE dynamics including inertial effects. The ISF has previously been derived [31] for a collection of point scatters undergoing rigid rotations described either by rotational jump diffusion, or continuous Brownian dynamics. However, just as the ISF for translational Langevin dynamics includes a ballistic/diffusive transition [32] distinguishing the dynamics from the spatial diffusion limit, the rotational dynamics should also exhibit a ballistic/diffusive crossover. In order to benchmark the implementation of rotational Langevin dynamics within PIGLE, we benchmark against an analytical result including the inertial effects that are both inevitable in the simulation, and in principle observable in experiment.

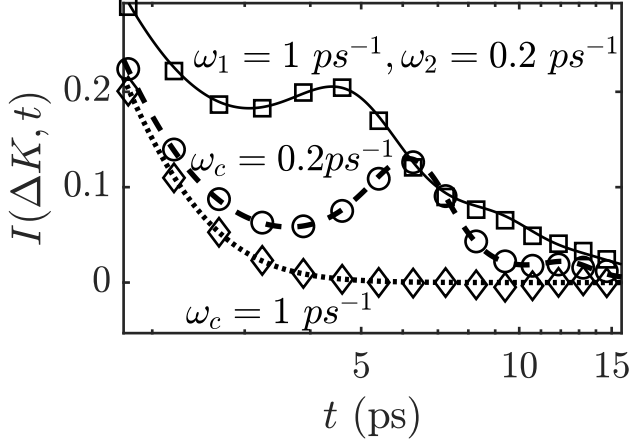


Figure 4: Benchmark PIGLE results (markers) with an analytic ISF (solid curves) for an isolated particle with mass of 104.15 amu on a flat PES. The ISFs were calculated for $\Delta K = 1 \text{ \AA}^{-1}$, and the particle obeys the GLE with a noise filtered with either an exponential (eq. 12) or a bi-exponential (eq. 14) low-pass filter (with global frictions of 1 ps^{-1} and 5 ps^{-1} , and corresponding decay constants of $\tau = 1 \text{ ps}$, $\tau = 5 \text{ ps}$).

We calculate the incoherent ISF associated with the Langevin dynamics of a single particle constrained to a circle of radius r , in the (x, y) plane. Its angular co-ordinate $\theta(t)$ is assumed to obey a Langevin equation

$$I\ddot{\theta} = -\gamma\dot{\theta} + G(t), \quad (17)$$

where I is the moment of inertia, γ is the rotational friction with units ps^{-1} , and $G(t)$ is a Gaussian zero-mean random torque analogous to the random force in the translational LE. The isotropy of the dynamics means we can choose $\Delta \mathbf{K}$ to point in an arbitrary direction, and we choose the x direction such that the scattering amplitude $A = \exp(i\Delta \mathbf{K} \cdot \mathbf{R})$ takes the form

$$A_{\Delta K}(\theta) = \exp \left[i\Delta K r \cos(\theta) \right]. \quad (18)$$

The ISF can then be expressed in terms of the initial probability $p(\theta)$ to find the molecule at an angle θ , and the conditional probability $G(\theta', \theta; t)$ for it to propagate to a new angle θ' in time t , by [31]

$$I(\Delta K, t) = \int_{-\infty}^{\infty} d\theta' \int_0^{2\pi} d\theta p(\theta) G(\theta', \theta; t) A_{\Delta K}(\theta') A_{\Delta K}(\theta). \quad (19)$$

For Brownian rotations, the conditional probability function $G(\theta', \theta; t)$ is given by the Green's function of a master equation describing spatial diffusion [31]. However, by analogy with the translational Langevin ISF [32], the short time ballistic behavior in Equation 17 is correctly accounted for by the solution

$$G(\theta', \theta; t) = \frac{\sqrt{I\gamma^2}}{\sqrt{4\pi k_B T [e^{-\gamma t} + \gamma t - 1]}} \exp \left(- \frac{I\gamma^2(\theta - \theta')^2}{4k_B T [e^{-\gamma t} + \gamma t - 1]} \right). \quad (20)$$

We apply the Jacobi-Anger expansion

$$\exp \left[i\Delta K r \cos(\phi) \right] = \sum_{n=-\infty}^{+\infty} i^n J_n(\Delta K r) e^{in\phi} \quad (21)$$

to both $A(\theta)$ and $A(\theta')$, after which the θ and θ' integrals produce the final result

$$I(\Delta K, t) = \sum_{n=-\infty}^{+\infty} J_n^2(\Delta K r) \exp \left\{ - \frac{k_B T n^2}{I\gamma^2} [e^{-\gamma t} + \gamma t - 1] \right\}. \quad (22)$$

Although in general helium scattering methods will measure the coherent ISF rather than the incoherent ISF, it is straightforward to show that the coherent rotational ISF for a homonuclear diatomic molecule is the same as the incoherent ISF in equation 22. The only change required to the calculation is to substitute the coherent scattering amplitude

$$A(\theta) = \frac{1}{2} \left[e^{i\Delta K r \cos(\theta)} + e^{-i\Delta K r \cos(\theta)} \right]. \quad (23)$$

Otherwise following through the evaluation of the general expression 19 as before, the result turns out to be the same because $J_n(\Delta K r) = J_{-n}(\Delta K r)$, although in general the coherent ISF will be different when the number of scattering centers changes.

Figure 5 shows benchmarking results for rotational motion, and primarily demonstrates that Equation (22) agrees with PIGLE simulation results of Langevin rotational motion in a flat potential energy landscape. To illustrate the angular inertial effects automatically present in the simulation, overlaid is the corresponding Brownian (spatial diffusion) result [31] that gives the same angular diffusion constant:

$$I(\Delta K, t) = \sum_{n=-\infty}^{+\infty} J_n^2(\Delta K r) \exp \left\{ - \frac{k_B T n^2}{I\gamma} t \right\}. \quad (24)$$

As in Brownian translational diffusion, the lack of inertial effects reveals itself as a cusp at the origin; the short time ballistic behavior in the simulations and in Equation 22 leads instead to smooth Gaussian behavior at short correlation times.

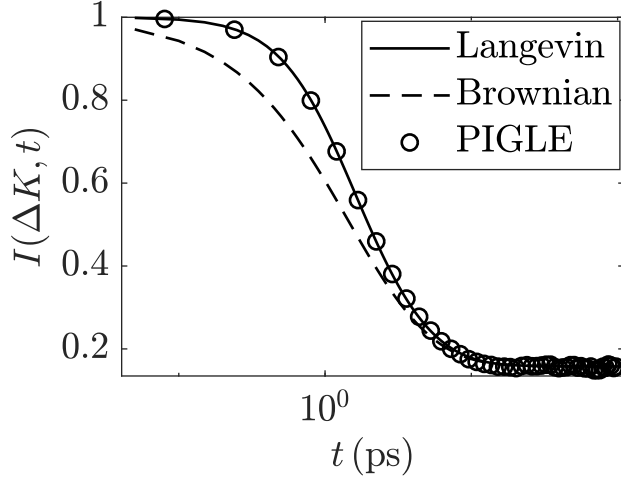


Figure 5: ISF calculated from PIGLE simulation for a diatomic molecule rotating on a flat PES (purple circles), is bench-marked against analytic ISFs for Langevin (solid blue) and 'pure' Brownian rotations (dashed red). The mass of the molecule is set to $m=104.15$ amu, with radius of $r = 4 \text{ \AA}$, friction of $\gamma = 2 \text{ ps}^{-1}$, at 300 K.

6.3. Potential Energy Surface and Interactions

To benchmark the implementation of the PES, we now make use of the Chudley-Elliott analytic model for single jumps on a Bravais lattice.[33] The model predicts exponentially decaying ISFs with a $\Delta\mathbf{K}$ -dependent decay rate called $\alpha(\Delta\mathbf{K})$, given the allowed site-to-site jump vectors and the corresponding jump rates. We chose to construct a PES for hexagonal surface, with several distinct high symmetry sites (commonly referred to as top, bridge, hcp hollow and fcc hollow). While the example PES is abstract, the symmetry and values we use are representative to diffusion of small molecules on (111) faces of transition metals. The PES was constructed using a Fourier series of the first six shells of G vectors (of a hexagonal lattice), with the Fourier coefficients calculated from values for six points in the unit cell (top, half way between top and hcp, hcp, half way between top and fcc, bridge, fcc). Let $V_{T,H,F,B}$ be the potential at the top, hcp, fcc, bridge sites. We use a

potential with $V_T = 0$ meV, $V_{H,F} = 130$ meV, and $V_B = 100$ meV. The half-way points between the top and two hollow sites were set to $V_{T-H,T-F} = 65$ meV (a function to create a similar PES is provided in the PIGLE package). The PES is effectively an array of 100 meV adsorption wells arranged in a Bravais lattice, and therefore, for a friction high enough to hinder multiple jumps, we can expect a qualitative agreement between ISF calculated from the simulation, and the Chudley Elliott model for single jumps. The simulation is performed with and without inter-adsorbate interactions enabled. The expected effect of inter-adsorbate interactions is described by de Gennes, [34] where a *dip* in the dispersion curve is expected at a momentum transfer proportionally inverse to the average distance of the adsorbates, which is governed by the number density.[7]

Figure 6 shows the decay rates of ISFs from a simulation of 50 interacting particles of mass 28 amu, at temperature of $T = 200$ K, and the PES described above. The friction was set to 6 ps^{-1} , with non-filtered random-impulse. The simulation time-step was 1 fs, and the total simulation time was 3 ns. The upper panels present results for a simulation with non-interacting particles. On two different azimuths the ISF was calculated for various momentum transfers, and the tail of the ISF, which is expected to hold information on the jump part of the particle's motion,[7] was fitted to a decaying exponent. The decay rates are plotted (circles), along with a fit to the Chudley Elliott model for single jumps (solid line).[33] As expected, the variation of the decay rates with $\Delta\mathbf{K}$ agrees with the Chudley-Elliott model for single jumps. The lower panels present results for a simulation of interacting particles, with the inter-adsorbate force defined to be $f(x) = -50 \cdot 10^4 \cdot x^{-4} \text{ meV/\AA}$. The inter-adsorbate interactions induce a de Gennes feature ([34]) at $\Delta\mathbf{K} = 0.45 \text{ \AA}^{-1}$, as expected for the number density (adsorbates per primitive unit cell) used in the simulation, 0.025. The unequal spacing in $\Delta\mathbf{K}$ is a consequence of the *in-phase* scattering condition in the simulation (see explanation above).

The calculation of the interacting system was performed using the high performance computing center at the University of Cambridge, on a node with 32 cores, and each iteration of 32 parallel runs took 28 minutes. A total of 96 simulation runs were performed to average the ISFs. The decay rates presented in Figure 6 were extracted by fitting the tail of the ISF to an exponential decay, separately for every $\Delta\mathbf{K}$ at which the ISF was computed.

A simulation of non-interacting adsorbates was used to validate the activation energy. For the same potential (with wells of 100 meV), simulations of

100 non-interacting particles were performed for temperature range between 150 K and 300 K. Extracting the incoherent ISF from each simulation is equivalent to running 100 simulations of a single particle. Figure 7 shows the natural logarithm of the decay rate of the ISF at $\Delta\mathbf{K} = 0.6 \text{ \AA}^{-1}$ along the $\langle 1, 1 \rangle$ high symmetry azimuth, against the temperature ($1/T$). The linear fit gives an activation energy of 83.25 meV which is slightly smaller than the PES well depth of 100 meV, as expected.[7]

7. Summary

PIGLE, a MATLAB/Simulink based package for molecular dynamics simulations, enables the expansion of the scope under which surface dynamics experiments are analyzed, within the Langevin framework. The main features of the simulator include a 4D PES (including rigid-body rotations), implementation of the Generalized Langevin Equation, and support in co-existence of multiple interacting species. In the paper we have presented the main building block of the simulator, and key information about procedures for post simulation analysis of the dynamical correlation functions of the simulated particles. Presented as well is the benchmarking of the rotational and translational motion, and of the generalized form of the Langevin Equation, which allows to apply more realistic noise spectrum to the surface fluctuations. Analytical models for the benchmarking are explained, including novel expressions for the intermediate scattering function of particles diffusing in a flat potential energy landscape. The analytical test cases are a diatomic molecule subject to Langevin rotational dynamics, and a point particle obeyed Generalized Langevin dynamics with bi-exponential memory friction.

8. Acknowledgements

This work was performed using the Darwin Supercomputer of the University of Cambridge High Performance Computing Service (<http://www.hpc.cam.ac.uk/>), provided by Dell Inc. using Strategic Research Infrastructure Funding from the Higher Education Funding Council for England and funding from the Science and Technology Facilities Council. For financial support, N.A. acknowledges the Blavatnik Foundation and the Herchel Smith fund.

References

- [1] Y.-G. Wang, D. Mei, V.-A. Glezakou, J. Li, R. Rousseau, Dynamic formation of single-atom catalytic active sites on ceria-supported gold nanoparticles, *Nature Communications* 6 (2015) 6511.
- [2] M. L. Klein, W. Shinoda, Large-scale molecular dynamics simulations of self-assembling systems, *Science* 321 (5890) (2008) 798–800.
- [3] G. Wang, A. Rühling, S. Amirjalayer, M. Knor, J. B. Ernst, C. Richter, H.-J. Gao, A. Timmer, H.-Y. Gao, N. L. Doltsinis, et al., Ballbot-type motion of n-heterocyclic carbenes on gold surfaces, *Nature Chemistry* 9 (2) (2017) 152.
- [4] J. Qian, W. A. Henderson, W. Xu, P. Bhattacharya, M. Engelhard, O. Borodin, J.-G. Zhang, High rate and stable cycling of lithium metal anode, *Nature Communications* 6 (2015) 6362.
- [5] M. Ma, F. Grey, L. Shen, M. Urbakh, S. Wu, J. Z. Liu, Y. Liu, Q. Zheng, Water transport inside carbon nanotubes mediated by phonon-induced oscillating friction, *Nature nanotechnology* 10 (8) (2015) 692.
- [6] G. Alexandrowicz, A. Jardine, Helium spin-echo spectroscopy: studying surface dynamics with ultra-high-energy resolution, *Journal of Physics: Condensed Matter* 19 (30) (2007) 305001.
- [7] A. Jardine, H. Hedgeland, G. Alexandrowicz, W. Allison, J. Ellis, Helium-3 spin-echo: Principles and application to dynamics at surfaces, *Progress in Surface Science* 84 (11-12) (2009) 323–379.
- [8] L. Chapman, D. Yoder, R. Colella, Virtual bragg scattering: a practical solution to the phase problem in diffraction, *Physical Review Letters* 46 (24) (1981) 1578.
- [9] A. Jardine, J. Ellis, W. Allison, Effects of resolution and friction in the interpretation of QHAS measurements, *The Journal of chemical physics* 120 (18) (2004) 8724–8733.
- [10] M. Diamant, S. Rahav, R. Ferrando, G. Alexandrowicz, Interpretation of surface diffusion data with Langevin simulations: a quantitative assessment, *Journal of Physics: Condensed Matter* 27 (12) (2015) 125008.

- [11] S. P. Rittmeyer, D. J. Ward, P. Gütlein, J. Ellis, W. Allison, K. Reuter, Energy dissipation during diffusion at metal surfaces: Disentangling the role of phonons versus electron-hole pairs, *Physical review letters* 117 (19) (2016) 196001.
- [12] P. Rotter, B. A. Lechner, A. Morherr, D. M. Chisnall, D. J. Ward, A. P. Jardine, J. Ellis, W. Allison, B. Eckhardt, G. Witte, Coupling between diffusion and orientation of pentacene molecules on an organic surface, *Nature materials* 15 (4) (2016) 397.
- [13] H. Hedgeland, M. Sacchi, P. Singh, A. J. McIntosh, A. P. Jardine, G. Alexandrowicz, D. J. Ward, S. J. Jenkins, W. Allison, J. Ellis, Mass transport in surface diffusion of van der waals bonded systems: Boosted by rotations?, *The journal of physical chemistry letters* 7 (23) (2016) 4819–4824.
- [14] E. Weinan, W. Ren, E. Vanden-Eijnden, String method for the study of rare events, *Physical Review B* 66 (5) (2002) 052301.
- [15] R. Elber, Long-timescale simulation methods, *Current opinion in structural biology* 15 (2) (2005) 151–156.
- [16] R. Kubo, The fluctuation-dissipation theorem, *Reports on progress in physics* 29 (1) (1966) 255.
- [17] J. Ellis, A. Graham, The use of quasielastic helium atom scattering to study correlated motion in adsorbate overlayers, *Surface science* 377 (1997) 833–842.
- [18] M. Nava, M. Ceriotti, C. Dryzun, M. Parrinello, Evaluating functions of positive-definite matrices using colored-noise thermostats, *Physical Review E* 89 (2) (2014) 023302.
- [19] P. S. M. Townsend, Diffusion of light adsorbates on transition metal surfaces, Ph.D. thesis, University of Cambridge (2018).
- [20] M. Ceriotti, A novel framework for enhanced molecular dynamics based on the generalized Langevin equation, Ph.D. thesis, ETH Zurich (2010).
- [21] J. Proakis, D. Manolakis, *Digital Signal Processing: principles, algorithms and applications*, 3rd Edition, Macmillan Publishing Company (New York), 1996.

- [22] M. Ceriotti, G. Bussi, M. Parrinello, Langevin equation with colored noise for constant-temperature molecular dynamics simulations, *Physical review letters* 102 (2) (2009) 020601.
- [23] SIMULINK, version 9.0, The MathWorks Inc., Natick, Massachusetts, 2017.
- [24] R. Collingham, Modelling molecular dynamics using the Generalised Langevin Equation, Part III Project Report, University of Cambridge (2016).
- [25] J. A. Anderson, C. D. Lorenz, A. Travesset, General purpose molecular dynamics simulations fully implemented on graphics processing units, *Journal of Computational Physics* 227 (10) (2008) 5342–5359.
- [26] M. Ghosh, A molecular dynamics approach to the surface diffusion of interacting adsorbates, Part III Project Report, University of Cambridge (2017).
- [27] G. H. Vineyard, Scattering of slow neutrons by a liquid, *Physical Review* 110 (5) (1958) 999.
- [28] L. Van Hove, Correlations in space and time and born approximation scattering in systems of interacting particles, *Physical Review* 95 (1) (1954) 249.
- [29] B. J. Berne, J. P. Boon, S. A. Rice, *J. Chem. Phys.* **45** (4) (1966) 1086–96.
- [30] P. S. Townsend, D. J. Ward, The intermediate scattering function for quasi-elastic scattering in the presence of memory friction, *Journal of Physics Communications* 2 (7) (2018) 075011.
- [31] I. Calvo-Almazán, S. Miret-Artés, P. Fouquet, *Journal of Physics: Condensed Matter* 24 (10) (2012) 104007.
- [32] S. Miret-Artés, E. Pollak, *J. Phys. Condens. Mat.* **17** (49) (2005) S4133–50.
- [33] C. T. Chudley, R. J. Elliott, Neutron scattering from a liquid on a jump diffusion model, *Proceedings of the Physical Society* 77 (2) (1961) 353.

- [34] P. de Gennes, Liquid dynamics and inelastic scattering of neutrons, *Physica* 25 (7-12) (1959) 825–839.

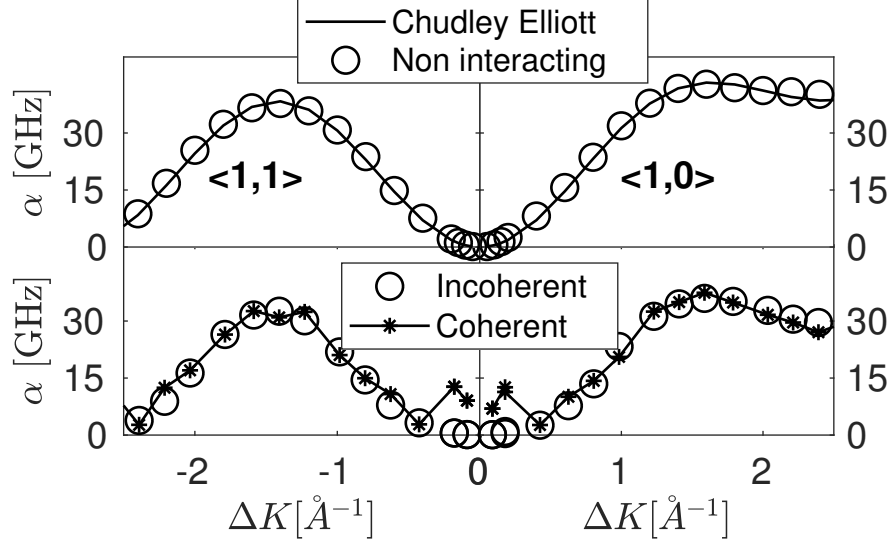


Figure 6: The effects of a corrugated potential and inter-adsorbate interactions are tested by following the changing decay rate of the ISF as function of momentum transfer. In the upper panels, simulation results for non-interacting particles are compared with the Chudley-Elliott analytic model for single jumps on a Bravais lattice (solid blue curves), for the two high symmetry azimuths of the underlying hexagonal lattice. The lower panels show results for a simulation with interacting particles; the curves show the variation of the decay rate of the incoherent (circles) and coherent (dots connected by a line) ISF calculated. The simulation parameters are described in the text. Left and right panels correspond to the two high symmetry azimuths of the underlying lattice (hexagonal surface with the lattice constant of copper crystal), $\langle 1,0 \rangle$ and $\langle 1,1 \rangle$. The non-interacting simulation agrees with a Chudley Elliott model for single jumps. Furthermore, the expected de-Gennes effect of inter-adsorbate interactions is well observed in the variation of the coherent ISF (lower panels), and its position ($\Delta K \approx 0.45 \text{ \AA}^{-1}$) matches the number density of the simulated adsorbates, 0.025.

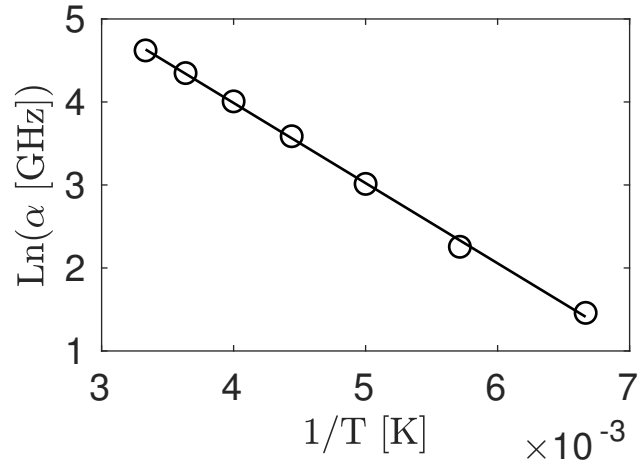


Figure 7: Arrhenius plot for decay rates at $\Delta K = 0.6 \text{ \AA}^{-1}$. The activation energy extracted from the linear fit is 83.25 meV, which is slightly smaller than the adiabatic barrier of 100 meV, as generally expected.[7].

Required Metadata

Current executable software version

Nr.	(executable) Software metadata description	Please fill in this column
S1	Current software version	1.0.0-beta2
S2	Permanent link to executables of this version	example: https://github.com/na364/PIGLE/releases/tag/v1.0.0-beta2 :
S3	Legal Software License	GNU/GPL-3.0
S4	Computing platform/Operating System	Any OS supported by MATLAB/Simulink, i.e. Linux, Microsoft Windows
S5	Installation requirements & dependencies	MATLAB/Simulink, with a compiler compatible with Simulink. See http : //mathworks.com/support/compilers.html)
S6	If available, link to user manual - if formally published include a reference to the publication in the reference list	
S7	Support email for questions	na364@cam.ac.uk

Table 1: Software metadata (optional)

Current code version

Nr.	Code metadata description	Please fill in this column
C1	Current code version	v1.0.0-beta2
C2	Permanent link to code/repository used of this code version	For example: https://github.com/na364/PIGLE/releases/tag/v1.0.0-beta2
C3	Legal Code License	GNU/GPL-3.0
C4	Code versioning system used	git
C5	Software code languages, tools, and services used	MATLAB/Simulink
C6	Compilation requirements, operating environments & dependencies	A compiler compatible with Simulink is required (See http://mathworks.com/support/compilers.html)
C7	If available Link to developer documentation/manual	
C8	Support email for questions	na364@cam.ac.uk

Table 2: Code metadata (mandatory)